

What is claimed is:

1. A software control interface for creating and editing system views of a business information system using object models shaped to define the system views comprising:

a drawing tool for visually representing system views using model elements represented by icons;

a first specification language for defining the various elements of a drawing; and

a second specification language for defining mapping between drawing icons and objects stored in an object repository of the software system;

characterized in that a user operating the drawing tool through a graphical user interface can, by dragging and dropping icons of the drawing tool onto a drawing sheet of the drawing tool, specify abstract models and model extensions expressed in diagrammatic notation transparent to the user and usable by the software system and associated repository.

2. The software control interface of claim 1 wherein the modeling system is defined according to a meta modeling framework that adheres to a hierarchical tri-level structure of abstraction.

3. The software control interface of claim 2 wherein the modeling framework includes a system information model, a meta model and a meta meta model.

4. The software control interface of claim 3 wherein the language for describing the system information model is UML.

5. The software control interface of claim 3 wherein the language for describing the system information model is XML.

5 6. A method for creating and manipulating business system views of a complex software system through user manipulation of visual symbols and connectors stored in an object repository associated with the system, the symbols and connectors represented in an interface of a drawing tool comprising steps of:

10 (a) providing a first specification language specifying the symbols and connectors of the drawing tool;

(b) providing a mapping second specification language for mapping drawing elements to repository objects;

15 (c) selecting certain symbols and connectors from a palette of the drawing tool, and dropping them in certain order on an open sheet of the drawing tool until a view or view edit is complete; and

(d) when complete, executing a completed system view for display on a graphical interface of the software system.

20 7. The method of claim 6 wherein in steps (a) through (d) a hierarchical tri-level meta modeling framework is observed.

25 8. The method of claim 7 wherein the meta modeling framework includes a system information model at a first level of abstraction, a meta model at a higher level of abstraction and a meta meta model at a highest level of abstraction.

9. An object modeling system for creating, editing, and displaying various views of a business information system comprising:

an object repository for storing object models and model elements;

a drawing tool for visually representing system views using model

5 elements represented by drawing icons associated with the tool;

a first specification language for defining the various elements of a drawing represented by the drawing icons; and,

a second specification language for defining mapping between the drawing icons and model elements stored in the object repository;

10 characterized in that the system operates according to a tri-level meta modeling framework including a meta meta model functioning as a base model for the instantiation hierarchy of the framework, a meta model formed as an instance of the meta meta model, the meta model defining the structure and semantics for an information system view, and an information
15 system model formed as an instance of the meta model, the information system model describing one or more specific information system views as specified by the meta model.

10. The object modeling system of claim 9 wherein UML, XML, and entity
20 relationship modeling language is supported at the highest level of abstraction.

11. The object modeling system of claim 9 wherein the icons comprise symbols and connectors.

25

12. The object modeling system of claim 9 wherein the association between the drawing icons and model elements stored in the repository is automated.

13. The object modeling system of claim 9 wherein dragging and dropping drawing icons from a drawing palette into a drawing sheet causes machine readable instruction for building system views in an object oriented way, the instruction code transparent to the user.

5

14. The object modeling system of claim 13 wherein the views are customizable and extendable over existing system views by creating new icons and connectors using the drawing tool, specifying them in the drawing and mapping languages and storing the elements in the object repository.

10

15

20

2025-10-10 10:10:10